

NAVAL POSTGRADUATE SCHOOL
Monterey, California



THESIS

**IMPROVING THE LAYOUT OF A
WAREHOUSE AT THE COAST GUARD
AIRCRAFT REPAIR AND SUPPLY
CENTER**

by

James F. Martin

September 1999

Advisor:
Second Reader:

Kevin R. Gue
Gordon H. Bradley

Approved for public release; distribution is unlimited.

DTIC QUALITY INSPECTED 4

19991126 101

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, Va 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September, 1999	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE IMPROVING THE LAYOUT OF A WAREHOUSE AT THE COAST GUARD AIRCRAFT REPAIR AND SUPPLY CENTER			5. FUNDING NUMBERS	
6. AUTHORS Martin, James F.				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT(maximum 200 words) We present a heuristic algorithm to evaluate alternative item and storage device locations in the Coast Guard's Aircraft Repair and Supply Center (ARSC) warehouse. The goal is to minimize the labor cost of item pickers by locating items in a way that reduces travel. The heuristic assigns items with the highest usage to the storage locations nearest the input/output point and evaluates alternative plans for relocating a limited number of storage devices by pairwise-interchange. We judge the quality of our results by comparing them to ARSC's current item locations and storage device layout. We also develop an iterative linear programming (LP) based algorithm that provides a lower bound on cost for comparison with the heuristic's results. Although implementing the iterative LP solution requires capital outlays beyond current budgets of the ARSC, the solution provides insight into layout and labor cost tradeoffs for long term planning. Our results show that expected travel distances and labor costs can be reduced by 40.2% by reassigning items to locations within ARSC's current configuration of storage devices. By interchanging only 7 of 51 storage devices ARSC could gain an additional 5.6% improvement for a total 45.8% reduction. Our iterative linear programming solution establishes a lower bound of 53.7% reduction over ARSC's current layout.				
14. SUBJECT TERMS Warehouse Layout, Order Batching, Aisle Travel Metric, Pairwise-Interchange, Assignment problem, Excel, VBA			15. NUMBER OF PAGES 62	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

Approved for public release; distribution is unlimited.

IMPROVING THE LAYOUT OF A WAREHOUSE AT THE COAST GUARD AIRCRAFT REPAIR AND SUPPLY CENTER

James F. Martin
Lieutenant Commander, United States Coast Guard
B.S., The United States Coast Guard Academy, 1986

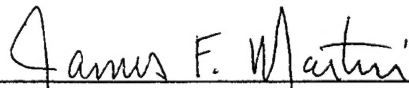
Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1999

Author:


James F. Martin

Approved by:



Kevin R. Gue, Advisor


Gordon H. Bradley, Second Reader



Richard E. Rosenthal, Chairman, Department of Operations
Research

ABSTRACT

We present a heuristic algorithm to evaluate alternative item and storage device locations in the Coast Guard's Aircraft Repair and Supply Center (ARSC) warehouse. The goal is to minimize the labor cost of item pickers by locating items in a way that reduces travel. The heuristic assigns items with the highest usage to the storage locations nearest the input/output point and evaluates alternative plans for relocating a limited number of storage devices by pairwise-interchange. We judge the quality of our results by comparing them to ARSC's current item locations and storage device layout. We also develop an iterative linear programming (LP) based algorithm that provides a lower bound on cost for comparison with the heuristic's results. Although implementing the iterative LP solution requires capital outlays beyond current budgets of the ARSC, the solution provides insight into layout and labor cost tradeoffs for long term planning. Our results show that expected travel distances and labor costs can be reduced by 40.2% by reassigning items to locations within ARSC's current configuration of storage devices. By interchanging only 7 of 51 storage devices ARSC could gain an additional 5.6% improvement for a total 45.8% reduction. Our iterative linear programming solution establishes a lower bound of 53.7% reduction over ARSC's current layout.

DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While an effort has been made within the time available to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

I.	INTRODUCTION	1
A.	THE AIRCRAFT REPAIR AND SUPPLY CENTER	1
B.	PROBLEM STATEMENT AND APPROACH	3
II.	LITERATURE REVIEW	5
A.	STORAGE POLICIES	5
B.	PRODUCT LOCATION IN A WAREHOUSE	7
1.	Distance in Aisles	8
2.	Distance Across Aisles	9
III.	AN ITEM LOCATION ASSIGNMENT ALGORITHM	11
A.	DEFINITIONS	11
B.	PROBLEM AREA AND DATA DESCRIPTION	12
C.	ASSUMPTIONS	15
D.	ALGORITHM	16
E.	RESULTS FOR THE CURRENT CONFIGURATION	18
IV.	RECONFIGURING STORAGE DEVICES	25
A.	A PAIRWISE-INTERCHANGE ALGORITHM	25
1.	Algorithm	25
2.	Results	27
B.	AN ITERATIVE LINEAR PROGRAMMING ALGORITHM	27
1.	Algorithm	27
2.	Results	30
V.	SUMMARY AND CONCLUSIONS	33
	APPENDIX A.	37
	LIST OF REFERENCES	41
	INITIAL DISTRIBUTION LIST	43

LIST OF FIGURES

1.	Warehouse Aisle Arrangement	8
2.	<i>Green</i> and <i>Yellow</i> storage device types	13
3.	Current ARSC Warehouse Layout	14
4.	Section A Issues By Row (Before)	19
5.	Section A Issues By Row (After)	19
6.	Section B Issues By Row (Before)	21
7.	Section B Issues By Row (After)	21
8.	Section E Issues By Row (Before)	22
9.	Section E Issues By Row (After)	22
10.	Section H Issues By Row (Before)	24
11.	Section H Issues By Row (After)	24
12.	Improved ARSC Warehouse Layout	28
13.	Best ARSC Warehouse Layout	32

LIST OF TABLES

1.	Storage Device Description	12
2.	Summary of Results	33

ACKNOWLEDGMENTS

This thesis is dedicated to my wife Tasha and our son Blake. They kept me going, maintained my balance and graciously gave more than they received. I would like to thank Professors Kevin R. Gue and Gordon H. Bradley for showing enthusiasm for this topic and all their hard work, dedication and support in helping me complete this thesis.

EXECUTIVE SUMMARY

Managers at the Coast Guard Aircraft Repair and Supply Center (ARSC) warehouse are developing ways to make warehouse operations cheaper and more efficient. They are anticipating an increase in the number of aircraft the Coast Guard operates with expanding Coast Guard aircraft operational requirements. More aircraft will require more warehouse activity with workers picking more parts to fill orders submitted by the Coast Guard's 24 Air Stations. Since it is unlikely that additional workers will be hired, managers desire to increase the efficiency of their current workforce by eliminating excessive and unnecessary travel throughout the warehouse. They also recognize that their current layout is inefficient. A more efficient workforce would allow for an increase in work requirements or the reprogramming of labor to other warehouse activities, such as inventory accuracy and housekeeping. Finally, the ARSC warehouse is near its capacity. Managers at ARSC desire to identify stagnant inventory for potential disposal or off site storage in order to free up warehouse space.

We seek to reduce labor costs at the ARSC warehouse by evaluating ways to reduce travel distances by relocating both items and storage devices. The ARSC warehouse presently does not employ any method of minimizing travel to pick orders for shipments. We address this problem by developing three different models to assign item and storage device locations and determine the fiscal feasibility of the solutions.

Our first model employs an algorithm that assigns an item to its optimal location within the current configuration of storage devices to minimize expected travel distance of picking items for an order. The model addresses two types of items: those that are picked individually and those that are picked in batches. We assign items that are picked individually to the location with the closest rectilinear distance to the input/output (I/O) point of the warehouse. For items that are picked in batches, we assume the aisle travel metric from the warehousing literature and assign items to the *aisle* closest to the I/O point. Total distance travelled is the sum of the

distance travelled to pick individual items and the expected distance travelled to pick the batched items. The algorithm also assigns the fastest moving items to storage locations that are physically easiest for the item pickers to reach. It also ensures that items that require more than one location are located in contiguous locations.

In our second model we determine an improved configuration of the storage devices (racks and shelving) by using a local search heuristic. Since managers are frequently limited by a budget when implementing changes, we determine the greatest reduction in expected travel distances within a specified budget. The heuristic makes limited adjustments to the current configuration by swapping the positions of storage devices along the main aisle.

Our last model is an iterative linear programming assignment algorithm in which we determine a lower bound on expected travel distance. Although implementation of a total redesign is not fiscally feasible, the lower bound solution provides insight into labor tradeoffs for long term planning.

Our results show that expected travel distances and labor costs can be reduced by 40.2% by reassigning items to locations within ARSC's current configuration of storage devices. By interchanging only 7 of 51 storage devices ARSC could gain an additional 5.6% improvement for a total 45.8% reduction. Our iterative linear programming solution establishes a lower bound of 53.7% reduction over ARSC's current layout.

Because total labor costs at ARSC are relatively small, the return on investment to relocate storage devices is not justified; but we argue that it is worthwhile to relocate items within the current configuration. Finally, we note that our methods are general enough to be applied to other warehouses in DOD or the distribution industry.

I. INTRODUCTION

Warehousing has been a part of civilization for thousands of years, initially as a means of storing foodstuffs during periods of abundance for use during times of scarcity. In the early days of the Industrial Revolution little thought was given to material handling practices because manpower was cheap and abundant. Modern warehousing practices emerged during World War II, primarily due to the increasing costs of labor and improvements in material handling devices (Ackerman, 1990). The modern warehouse has been evolving ever since. A successful warehouse operation requires effective use of space, equipment, and labor as well as accessibility and protection of inventory. Tompkins and Smith (1988) estimate that approximately fifty percent of warehouse costs are labor related. Reducing the amount of labor as well as increasing labor productivity can greatly reduce the costs of operating a warehouse.

A. THE AIRCRAFT REPAIR AND SUPPLY CENTER

The Coast Guard's Aircraft Repair and Supply Center (ARSC) is the inventory control point and primary repair facility for the Coast Guard's fixed and rotary wing fleet of aircraft. The Aviation Supply Division of ARSC's mission includes the receipt, storage, issue, and shipment of aircraft parts to the Coast Guard's 24 Air Stations throughout the United States, Puerto Rico, Alaska, and Hawaii. The warehouse covers approximately three-acres in Elizabeth City, NC.

The warehouse stores more than 39,000 line items of inventory equating to approximately 5.5 million individual parts; the inventory is valued at more than \$385M. The warehouse fills 110,000 requisitions and processes 66,000 receipts annually. More than 250 items are shipped to the Air Stations daily, with an additional 50 or more items shipped daily to the Repair Division of ARSC (Burgess, USCG, 1998). The Aviation Supply Division essentially has cradle-to-grave responsibility for the Coast Guard's aviation inventory.

During the early 1990's, ARSC underwent a major change in the way it controlled and stored matériel, with the adoption of the Aviation Matériel Management and Information System (AMMIS). Under the old system, items that had the same form, fit, and function were controlled and stored together. Under AMMIS, items are stored and controlled by individual National Stock Number (NSN). In order to improve inventory accuracy, managers at ARSC undertook a total rewarehousing effort in the early 1990's in which items were identified, marked, and relocated with the goal of achieving one NSN per location. This rewarehousing had the effect of significantly increasing the number of required storage locations since related items are no longer stored together. The increase in the number of locations required has pushed the warehouse location requirements close to its capacity.

The ARSC warehouse has evolved into a system of dedicated storage in which each NSN has a specific storage location, set of locations, or shared location. The inventory at ARSC is dedicated in that each item is stored in a dedicated location on open-face shelving, drawers, or bulk storage locations; but the locations of the items are *randomly* distributed throughout the warehouse, meaning that a frequently picked item is equally likely to be located near or far from the I/O point.

Managers at ARSC are also concerned with space availability in the warehouse. They desire to identify stagnant stock for potential disposal or off site storage in order to free up warehouse space. Additionally, they expect an increase in the number of aircraft the Coast Guard operates, although the aircraft types will not change. With more aircraft, there will be more trips workers must make to pick items in the warehouse. Since it is unlikely that additional workers will be hired, managers desire to increase the efficiency of their current workforce by eliminating excessive and unnecessary travel throughout the warehouse. ARSC desires to make each trip to pick parts as efficient as possible. Man-hours are currently being lost by ARSC's eight item pickers in unnecessary transit to the far reaches of the warehouse.

B. PROBLEM STATEMENT AND APPROACH

Reducing travel time reduces the labor costs of operating the warehouse, or it allows for the allocation of labor to other activities. Our problem is to determine how to relocate items within the warehouse and relocate the racks in which the items are stored in order to reduce travel distance. We develop three approaches to our problem: The first is to assign items to locations within ARSC's current layout to minimize expected travel; the second is to identify a limited number of storage racks, which, if relocated, would yield significant travel savings; the third is to find the best configuration of racks for the entire warehouse. We develop a model for each approach.

Estimating the cost of a given device configuration and item assignment for forklift items is straightforward, because data from ARSC suggests that only one item is picked per trip: we compute the distance to each item and multiply by the cost per foot of travel. (We estimate the cost coefficient with time study standards and current labor rates.) For the remaining items, which are picked in batches, we develop a model to estimate total travel per batch.

Our second model determines an improved configuration for the storage devices. Because ARSC is only willing to make limited adjustments to the current configuration, we find those adjustments that yield the greatest reduction in labor cost. We use a pairwise-interchange algorithm as a means to limit the number of devices requiring movement.

Our final model determines the best device layout in order to understand what gains would be possible if ARSC could undertake a significant warehouse redesign.

The rest of the thesis is organized as follows. Chapter II provides a literature review on warehouse storage policies and summarizes a stochastic model for optimal product layout in a warehouse. Chapter III provides a description of the ARSC warehouse, defines terms, establishes our assumptions, and describes the first model, the Item Location Assignment Algorithm. Chapter IV describes the second

and third models, the Pairwise-interchange Algorithm and the Iterative Linear Programming Algorithm respectively, which both deal with reconfiguring the storage devices. Finally, Chapter V summarizes the thesis and makes our implementation recommendation to ARSC.

II. LITERATURE REVIEW

Warehouse design includes determining the best warehouse location, layout, storage methods, equipment and automated systems, item location, zones, and order picking methods (Ashayeri and Gelders, 1985). Gray et al. (1992) reported that solving the entire warehouse design and operation problem "formally with a monolithic model appears impractical even in a small case." We address the problem of item and storage device location within ARSC's existing warehouse, using its existing storage devices.

The three main methods used to solve warehousing problems are analytic methods, simulation methods, and heuristic methods. Ashayeri and Gelders (1985) suggest that the best approach is a combination of analytic and simulation methods. We use analytic methods to optimize item location, a heuristic to improve storage device locations, and an iterative linear program to determine the best storage device locations and minimal expected travel distances.

A. STORAGE POLICIES

A storage *policy* determines the location and manner in which items are stored in a warehouse. There are three main alternative storage policies: random, dedicated, and class based storage.

Random storage occurs when an incoming item is assigned randomly to any location that is currently empty, usually the empty location that is closest to the Input/Output (I/O) point. The advantage to the policy is that it requires less storage space than the dedicated policy (Heragu, 1997). Francis et al. (1992) report that random storage requires a "good locator system to keep track of the location of products; with dedicated storage you know the permanent address of the product." Time will be wasted if there is not a good locator system when using random storage. Also, with dedicated storage the "fast movers" are assigned the locations closest

to the I/O point and the "slow movers" are assigned the farthest locations. With random storage a close location might not be available when a fast mover arrives forcing it to be stored in an unfavorable location. The same is true for a slow mover; it may arrive at a time when the next available slot is a highly desirable location. These results lead to greater expected travel distances with random storage versus dedicated storage (Francis et al., 1992).

In industry, there are two common variations of dedicated storage. The first is to store items in part number sequence. The other is to store items by some index related to their characteristics. Items are typically ranked by their index and stored relative to the I/O point by their rank (Guenov and Raeside, 1992). The most common form of dedicated storage uses Heskett's (1963) Cube-Per-Order Index (COI), an application of which is summarized by Kallina and Lynn (1976). The COI for an item is the space required for the item divided by its order frequency. Items are assigned locations in increasing order of COI, with the lower COI's placed closer to the I/O point (Wilson, 1977). Harmatuck (1976) showed that the COI rule is optimal for the class of problems in which an "out and back" selection method is used (either single- or dual-command systems) and when the cost of moving all items is constant and proportional to the distance travelled (Cormier and Gunn, 1992).

The third most common storage policy is class-based storage. Heragu (1997) describes a class-based dedicated storage policy that takes advantage of the Pareto Effect in that in a warehouse, 80% of the shipping and receiving activity is typically directed at 20% of the items, 15% at 30% of the items, and the remaining 5% of the activity is attributable to 50% of the items. These items can be classified as "fast movers," "medium movers," and "slow movers" (Nelson, 1985). To minimize the amount of time items spend being retrieved; the fast movers should be located closest to the I/O point; the medium movers should be located the next closest, and the slow movers should be located the farthest from the I/O point (Heragu, 1997). Hausman et al. (1976) compared the performance of the three systems in

an automated warehousing system by measuring crane travel times. They found that class based turnover storage assignment policies provided the most significant reduction in crane travel times.

Other guidelines for storage policies include the similarity philosophy which stores together items that are commonly ordered together and the size philosophy which recommends storing bulky, hard to move items closest to the I/O point (Hudock, 1996).

B. PRODUCT LOCATION IN A WAREHOUSE

Jarvis and McDowell (1991) develop a stochastic model for locating products in an order picking warehouse that minimizes the time to pick items in an order. They show that a simple assignment algorithm optimally assigns items to locations. They also develop a mathematical model to minimize the expected distance travelled per order. Because their work is essential to our problem, we summarize it here.

Jarvis and McDowell consider a rectangular warehouse illustrated in Figure 1, with the picking aisles perpendicular to the I/O point and with cross aisles at the aisle ends. The m picking aisles all have equal length l . A picking tour begins at the I/O point.

Jarvis and McDowell simplify the problem by considering half a symmetric warehouse (the I/O point is located adjacent to the first aisle and all aisles are located to the left of the I/O point as the worker looks out from that point). The item picker begins a tour by travelling to the first aisle containing an item on the order. The picker proceeds to the left most aisle containing an item on the order and only enters those aisles which contain items on the order. The authors assume that if an aisle is entered, it is completely traversed. The picker ends the tour by returning to the I/O point.

In Jarvis and McDowell's model, the total distance traveled (D_T) for any order is the sum of the distance travelled in aisles (D_I) and the distance travelled to the

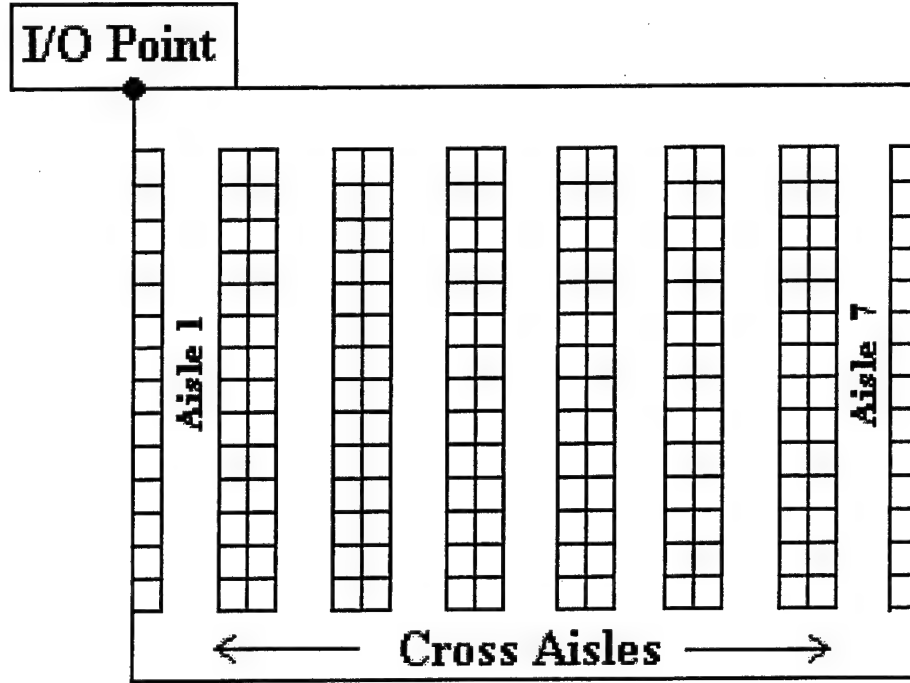


Figure 1. Aisle arrangement.

left of the I/O point (D_L). Therefore, the expected distance travelled per tour is $E(D_T) = E(D_I) + E(D_L)$.

1. Distance in Aisles

Because once an aisle is entered it must be completely traversed, $E(D_I) = lE(\text{number of aisles entered}) = lm - lE(\text{number of aisles not entered})$. Let p_i represent the probability that the i th product is on an order, $q_i = 1 - p_i$, and $A_j = \{i \mid \text{product } i \in \text{aisle } j\}$. Then the probability that none of the products on the j th aisle are on an order is $Q_j = \prod_{i \in A_j} q_i$, and

$$E(\text{number of aisles not entered}) = \sum_{j=1}^m Q_j.$$

Thus,

$$E(D_I) = l \left(m - \sum_{j=1}^m Q_j \right)$$

and minimizing $E(D_I)$ is equivalent to maximizing $\sum_{j=1}^m Q_j$.

This leads to the following corollary:

Corollary 1 (*Jarvis and McDowell*) *Expected in-aisle travel is minimized for the given warehouse by grouping the products with the lowest demand in an aisle, the products with the next lowest demand in another aisle, and so on, so that the products with the highest demand are grouped in an aisle.*

Note that Corollary 1 assumes independence of items on orders. We do not have data detailed enough to do correlation analysis; therefore, we assume independence as well.

2. Distance Across Aisles

To pick an order, the worker must visit each aisle that contains an item on the order. The minimum round trip distance D_L the worker travels to the left of the I/O point is twice the distance from the I/O point to the leftmost aisle picked. If we number the aisles such that $0 \leq d_1 < d_2 < \dots < d_m$, where d_j represents the distance from the dock to the j th aisle then,

$$E(D_L) = \sum_{j=1}^m d_j P(j\text{th aisle is the furthest picked}).$$

The probability that the j th aisle is the furthest picked is the probability that none of the items located in aisles beyond aisle j are on the order and at least one item on the order is located in aisle j . Let $P_j = 1 - Q_j$ be the probability that at least one item on the order is in aisle j .

Then,

$$\begin{aligned} P(j\text{th aisle is the furthest picked}) &= P_j \prod_{i=j+1}^m Q_i \\ &= (1 - Q_j) \prod_{i=j+1}^m Q_i, \end{aligned}$$

and

$$E(D_L) = \sum_{j=1}^m d_j (1 - Q_j) \prod_{i=j+1}^m Q_i,$$

where $Q_{m+1} = 1$.

This leads to the following corollary:

Corollary 2 (*Jarvis and McDowell*) *Assigning the least frequently picked products (largest q_i) to the leftmost distant aisle, the next least frequently picked items to the next most distant aisle and so on until the most frequently picked items are assigned to the aisle nearest the I/O point minimizes travel across aisles to the left of the I/O point.*

Since the I/O point of the ARSC warehouse is located near the end of all the aisles in the warehouse, we consider it to be “half a symmetric warehouse.” Therefore, the ARSC warehouse fits nicely into the type of warehouse Jarvis and McDowell consider in their model. Their model provides the framework for the model we use to compute expected travel distances of the ARSC workers for items that are picked in batches.

III. AN ITEM LOCATION ASSIGNMENT ALGORITHM

Warehousing problems are well known and researched throughout the operations research, industrial engineering, and management science fields (Heragu, 1997). Many mathematical models, algorithms and simulations have been devised to address these problems. We employ an Item Location Assignment Algorithm that assigns an item to its optimal location within the warehouse's configuration to minimize travel distance.

A. DEFINITIONS

Before proceeding with our discussion, we first define a number of terms.

- **order:** An item or set of items that will be picked on a single tour of the warehouse.
- **bin:** The shelf space where an item is stored.
- **bay:** A column of bins stacked one above another having equal widths.
- **row:** A set of adjacent bays that run the length of an aisle.
- **pick face:** The vertical face of a row. It includes all the bays and bins in a row.
- **rack:** A pair of rows that are placed back to back. Their pick faces are in adjacent aisles.
- **location:** A three part, six digit alpha-numeric code that describes the physical location of an item in the ARSC warehouse. For example, H1105B indicates the location in section H, row 11, bay 5 and bin B. The rows are numbered sequentially in each section with the first row being at the back of the section with respect to the I/O point. The bays are numbered sequentially with respect to the warehouse's main aisle. The bins in each bay start with the letter "A", which is the bin highest above the floor.
- **storage device type:** The ARSC warehouse has rows of six different dimensions, each of which we consider to be a different type which we define by a

Device Type	Bin Size	Bins per Bay	Shelves per Bay	Means of Retrieval
<i>Green</i>	3×2	Various	Various	Electric Cart
<i>Yellow</i>	8×4	6	3	Electric Cart
<i>Orange</i>	9×4	8	4	Electric Cart
<i>Blue</i>	12×4	6	3	Forklift
<i>Pink</i>	9×4	6	3	Forklift
<i>Brown</i>	14×4	6	3	Forklift

Table 1. Storage Device Description. The Bin Size is width times depth in square feet. For all device types except *Green* there are two adjacent bins per shelf.

color. A description of the different device types is summarized in Table 1. Figure 2 illustrates *green* and *yellow* device types.

- **rectilinear distance:** Rectilinear distances are measured using the formula $d_R = |x_1 - x_2| + |y_1 - y_2|$, where (x_1, y_1) are the coordinates of the I/O point and (x_2, y_2) are the coordinates of a location. This means that travel from the I/O point and an item location occurs along only one Cartesian axis at a time (Gibson and Sharp, 1992).
- **aisle travel metric:** The aisle travel metric characterizes the distances along the cross aisles in a warehouse similar to that at ARSC (see Figure 1). We assume that once an aisle is entered it must be completely traversed. Thus, the distance between items in different aisles is the distance between the centerlines of those two aisles. The distance between two items in the same aisle is defined to be zero.

Figure 3 shows the current warehouse layout with the storage device types color coded. The current layout has the benefit of uniformity of device types within sections which enhances worker familiarity.

B. PROBLEM AREA AND DATA DESCRIPTION

At ARSC's request, we are concerned only with storage devices in sections A, B, E, and H. We do not address bulk storage locations because managers at ARSC stated that movement of bulk items and the Vidmar (trademark) cabinets in Section I is not feasible. Sections C and F are the stocking locations for repairable items that

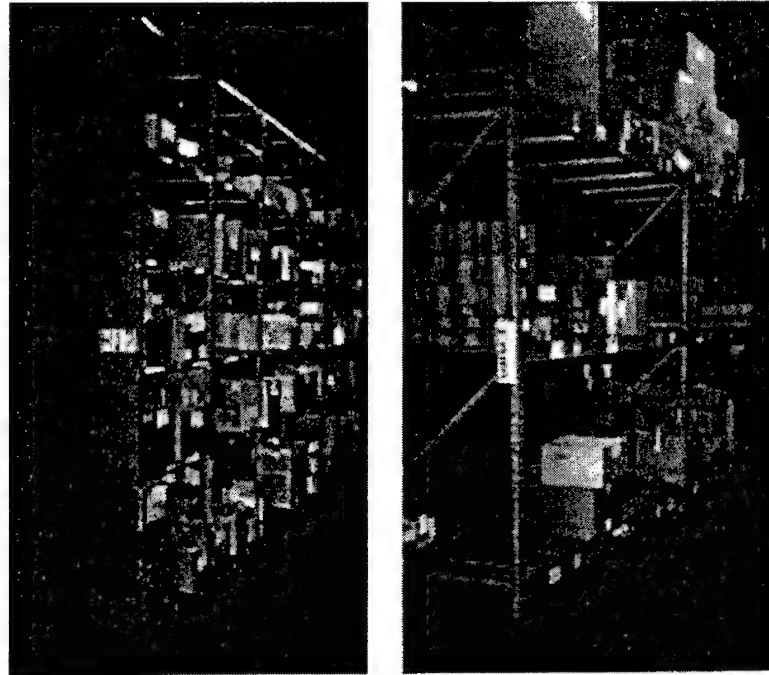


Figure 2. A *green* storage device type is depicted above left. Each shelf denotes a bin; the number of bins in a bay is variable. A *yellow* storage device type is depicted above right. Note that each *yellow* bay contains three shelves. Each shelf contains two adjacent bins.

are awaiting batch repair and are not shipped on a regular basis. These requirements reduce our problem to assigning 15,771 items within Sections A, B, E and H.

We extracted data from the AMMIS database and warehouse blueprints. We use data from the warehouse blueprints to determine distances along the warehouse's main aisle to the pick face for each row. The AMMIS database is in spreadsheet format and includes the item stock number, the current warehouse location of every item, and the number of issue events for an item over the five calendar quarters from 1 January 1998 to 31 March 1999. An issue event consists of a visit to a location in order to pick an item to fill an order. Additionally, ARSC provided the following data:

- The travel speed of the electric cart for batch picks is 8.57 feet per second; the travel speed of the forklift is 6.0 feet per second.
- The cost to reconfigure the entire warehouse is more than \$800,000.

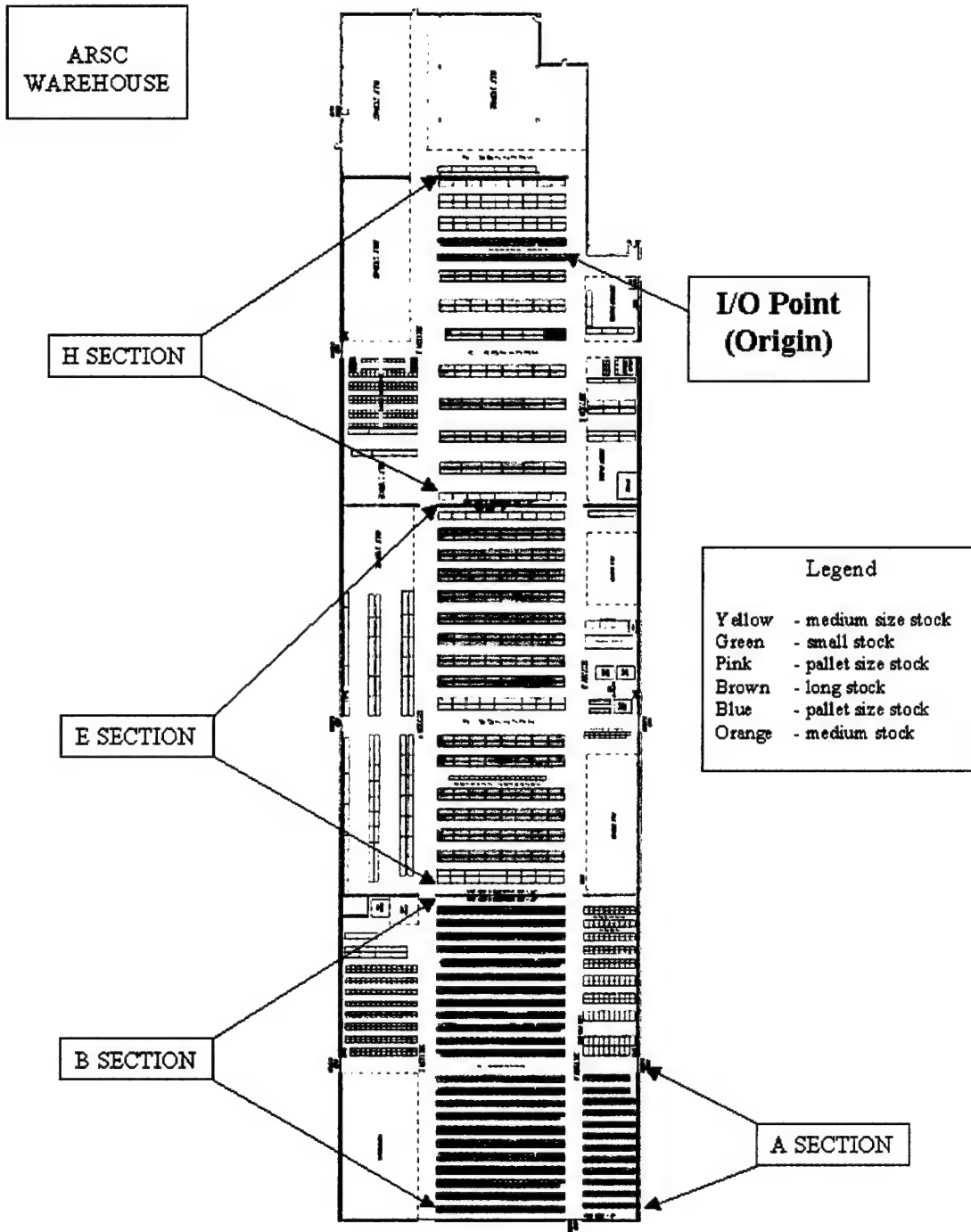


Figure 3. The current layout of the ARSC warehouse with storage devices color coded.

- The cost to reconfigure 10% of the storage devices is approximately \$100,000.
- The average hourly wage for item pickers is \$14.28 per worker for 8 workers.
- *Green* device locations store two items each (Sections A & B and rows H16–H19). All other locations store a single item.
- The stock number and number of locations required for storage are provided for 22 items that require more than one location.

The last item raises an important issue. Our problem becomes much more difficult if multiple units of an item must be stored contiguously in an aisle. In fact, the algorithm we develop for assigning items to locations assumes that an item occupies only one location. Because only 22 of 15,771 items require multiple (contiguous) locations, we assume that items occupy only one location and deal with these 22 items as special cases within our algorithm.

C. ASSUMPTIONS

We make the following assumptions, all of which are considered reasonable by managers at ARSC (McCarty and Limbacher, USCG, 1999):

- Workers pick
 - one forklift item per trip (*blue*, *pink*, and *brown devices*),
 - 10 large batch pick items per trip (*yellow* and *orange devices*), or
 - 50 small batch pick items per trip (*green devices*).
- An item must remain stored in the same type of device (color code) as it was originally found.
- Distance traveled for forklift picks is deterministic; distance traveled for batch picks is stochastic.
- Acceleration and deceleration time of a travel vehicle is negligible.
- The time to pick a product off a shelf is constant for forklift locations and batch pick locations.
- Aisle widths for forklift picks allow the vehicle to turn around and return while remaining within the aisle.

- For batch picked items we assume that once an aisle is entered the entire aisle is traversed.
- Small batch pick items (*green* devices) have storage bays with varying number of bins. We assume that all (*green*) items can fit in any (*green*) bin.

D. ALGORITHM

The ARSC warehouse stores two types of items, forklift items and items picked in batches. We compute travel distances for each type separately. Since forklift items are picked one at a time, the algorithm assigns highest usage items to the closest locations from the I/O point. The round trip distance travelled by all the forklift picks is,

$$D_F = \sum_{l=1}^k (2)(\text{Rectilinear Distance})_l (\text{Number of Issues})_i$$

where k is the number of locations with items requiring a forklift pick and i denotes the item stored in location l .

For batch picked items the algorithm assigns the highest usage items to the closest location in the closest available aisle to the I/O point. There will be cases when the item picker enters an odd number of aisles and ends his tour on the opposite side of the warehouse with respect to the I/O point. When this happens the picker must traverse an extra aisle to return to the I/O point. Jarvis and McDowell note that it is difficult to account for this additional aisle. We account for this additional aisle by first noting that 1.2% of the ARSC warehouse's items account for 72.8% of all the issue activity within Sections A, B, E, and H. When picking an order one of two possible cases will occur. In the first case the item picker traverses exactly one aisle. Let p_1^* be the probability that the *first* aisle is the only aisle picked,

$$p_1^* = (1 - Q_1) \prod_{i=2}^m Q_i.$$

If the first aisle is the only aisle picked then the in-aisle travel distance is,

$$E(D_I) = 2lp_1^*.$$

In the second case the item picker must traverse more than one aisle and the possibility exists that he will end his tour on the opposite side of the warehouse with respect to the I/O point. We assume that this will happen 50% of the time. If more than one aisle is picked then the in-aisle travel distance is

$$E(D_I) = (1 - p_1^*)l[m - \sum_{j=2}^m Q_j + 0.5].$$

Combining the two possible cases is the total expected in-aisle travel distance,

$$E(D_I) = 2lp_1^* + (1 - p_1^*)l[m - \sum_{j=2}^m Q_j + 0.5].$$

Therefore, the total expected distance travelled for a warehouse configuration is

$$D_T = D_F + \sum_{i \in B} b_i(E(D_I) + 2E(D_L)),$$

where B is the set of devices requiring batch picks and b_i is the number of batches picked per time for device type i .

The algorithm ensures that all A & B bins in Sections A and B and in rows H16-H19 (*green* devices) are left vacant due to difficulty in reaching them (they are high above ground). The algorithm also ensures that all A & B bins in Sections E and H (all non-*green* devices) are only filled with the slowest moving items. The algorithm first fills the lower bins with the high usage items. Once all the low, or "prime", locations have been filled with the highest usage items, the A & B bins are filled with what is left over. The A & B bins are high off the floor, requiring an uncomfortable and potentially hazardous overhead reach by a worker in order to pick an item. Our method keeps the high usage items "close to the floor and close to the door" while ensuring that the fast moving items are in an easy to reach location, providing for a "safe" pick.

There are 22 items that require multiple locations. The algorithm assigns all the units for any of these 22 items to locations in the same aisle and in contiguous bays. When the algorithm encounters one of these items, it determines if enough bays

remain available in the aisle to assign all the items. If so, it makes the assignment. If not, it skips to the next closest aisle, makes the assignment, and backfills the skipped locations with the next highest usage items. This ensures that workers only transit to one area of contiguous locations in the warehouse to pick an item rather than visiting several noncontiguous locations that may be temporarily out of items.

The Item Location Assignment Algorithm assigns each item to its optimal location within a given warehouse configuration subject to the constraints noted above. Pseudo-code for the algorithm is in Appendix A.

We implement our algorithm in Microsoft Excel 97 with Visual Basic Applications (VBA). Solutions for the assignment algorithm take approximately 6 minutes on a PC with a Cyrix 6x86 P200+ microprocessor (Intel 200 Mhz Pentium Pro processor equivalent).

E. RESULTS FOR THE CURRENT CONFIGURATION

We graph the issue events for each section before and after running the Item Location Assignment Algorithm. Figure 4 depicts the graph of Section A under the warehouse's current operating situation. This graph illustrates the nearly uniform manner in which items are currently distributed throughout the section. When we apply the Item Location Assignment Algorithm, we see in Figure 5 that Section A is now stocked with only stagnant inventory. Section A consists of *green* devices only. The fast moving stock previously located in Section A is reassigned to *green* devices in other sections that are closer to the I/O point.

Figure 6 depicts the graph of Section B with rows H16–H19 (*green* devices) before we reassign item locations. We include rows H16–H19 because they are the only rows of the *green* device type that are not in Section A or B. They are the *green* rows that are located closest to the I/O point and depicted on the far right side of Figures 6 and 7. With the exception of rows H16–H19, we see that issue events are nearly uniformly distributed throughout the section and without regard to the

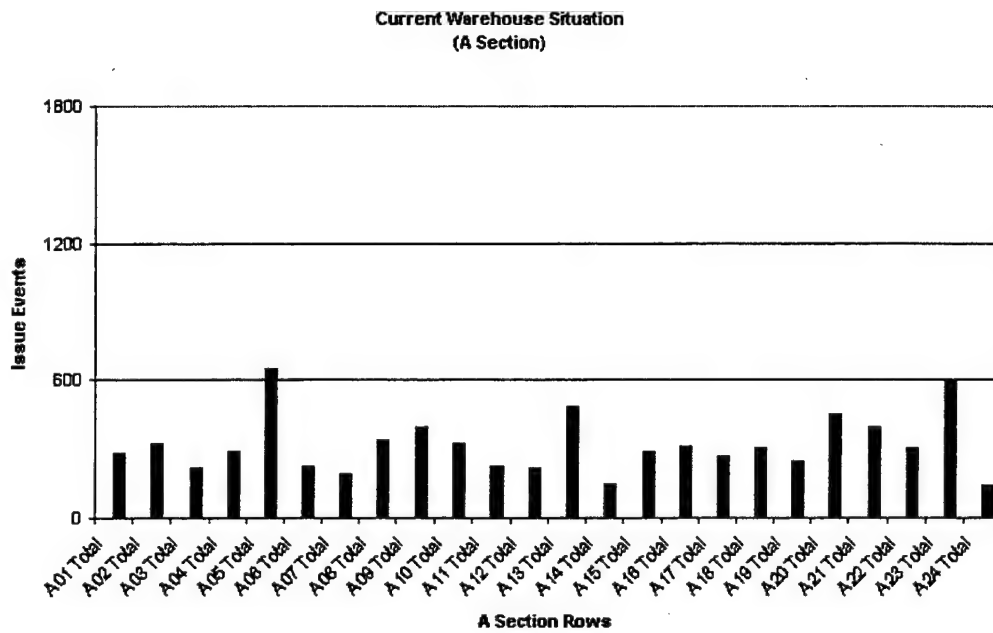


Figure 4. Section A issues by row before stock relocation

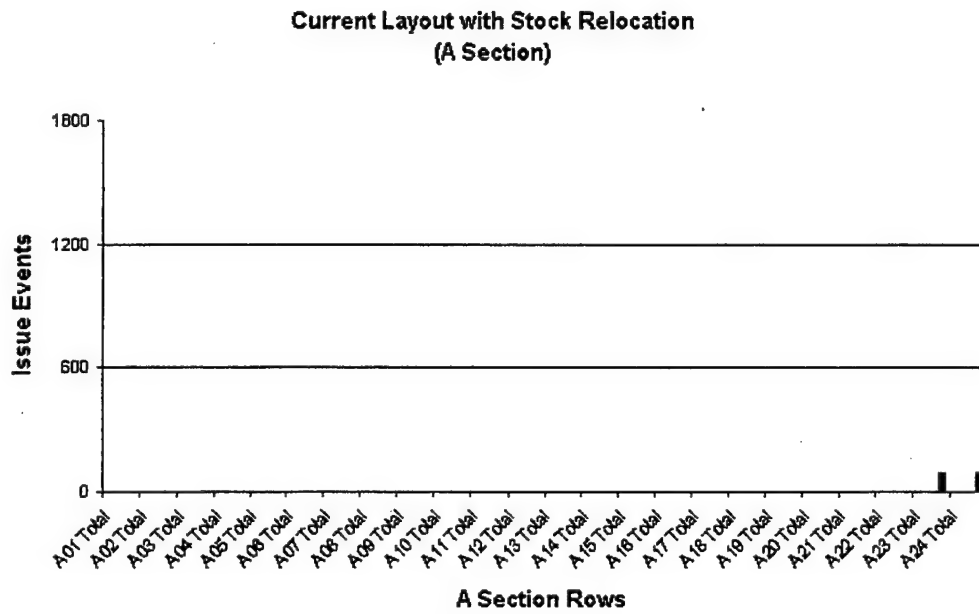


Figure 5. Section A issues by row after stock relocation

proximity of items to the I/O point. The greater number of issue events observed for rows H16–H19 are the result of initial attempts by ARSC to locate fast moving stock close to the I/O point. When we apply the Item Location Assignment Algorithm, Section B is stocked with the fast moving items located in the rows closest to the I/O point (Figure 7). Note that the issue events for rows H16–H19 have increased dramatically and that the back of the warehouse (rows B01–B20) contains purely stagnant inventory. The increase in issue events at row B36 is due to an increase in the number of bins per bay allowing more items to be stocked in that row; so rows B37 and greater have fewer bins. Figure 7 otherwise depicts a steady decrease in the number of issue events as we get farther from the I/O point.

Figures 8 and 9 depict the graphs of Section E. The rows in this section contain only *orange* and *yellow* devices. Prior to applying the Item Location Assignment Algorithm, we again observe the haphazard distribution of items depicted in Figure 8. Figure 9 depicts the results of relocating the fast movers to the forward rows. Note that the slight inconsistencies to an otherwise smooth curve are the result of *yellow* devices in rows E33, E16, E15, E02, and E01 which make up their own independent curve. The Section is predominantly made up of *orange* devices which otherwise smoothly decrease to stagnant rows as we move to the back of the warehouse away from the I/O point. Figure 9 depicts a steady decrease in the number of issue events as rows get further from the I/O point for both the predominant *orange* devices and the *yellow* devices.

The graphs of Section H are more subtle due to the number of different device types located there. Section H contains predominantly *pink*, *blue*, and *brown* devices picked by forklift but also contains a few *orange*, *yellow*, and *green* rows picked in batches. Additionally, we use the rectilinear metric instead of the aisle metric for the *pink*, *blue*, and *brown* devices since those items are retrieved by forklift one at a time. We have omitted rows H16–H19 from Figure 10 and Figure 11 (these *green* rows are depicted in the Section B graphs). Despite the wide variety of device types located

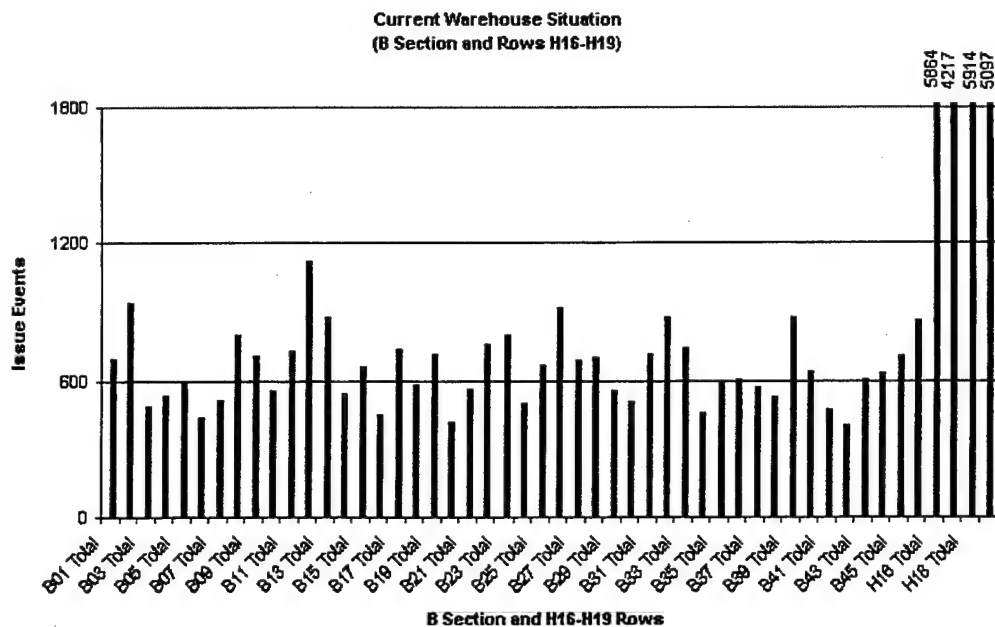


Figure 6. Section B issues and Rows H16-H19 issues by row before stock relocation with the Item Location Assignment Algorithm

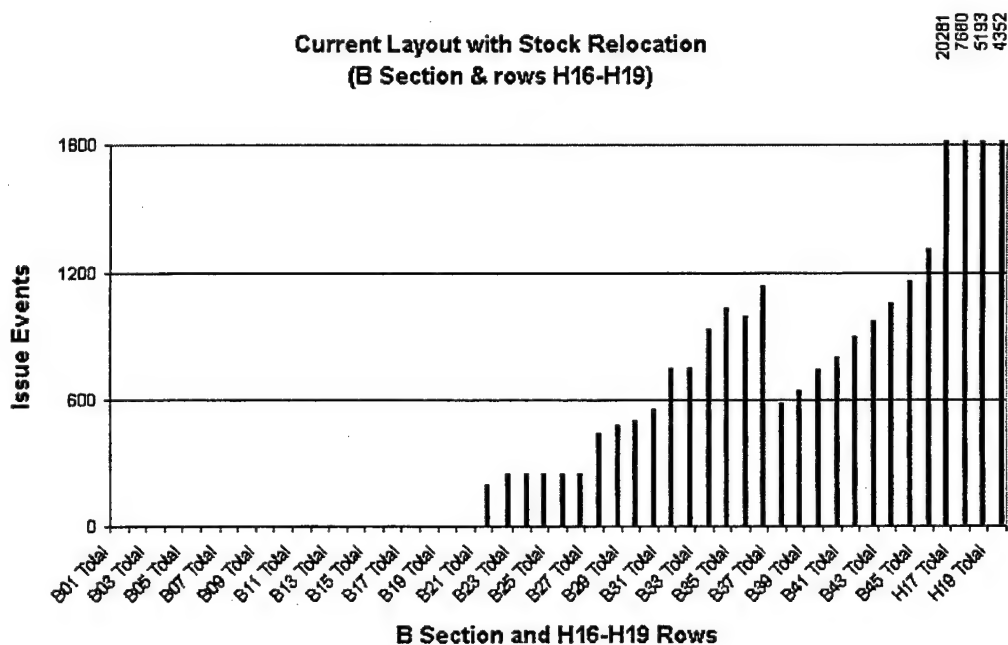


Figure 7. Section B issues and Rows H16-H19 issues by row after relocating stock with the Item Location Assignment Algorithm

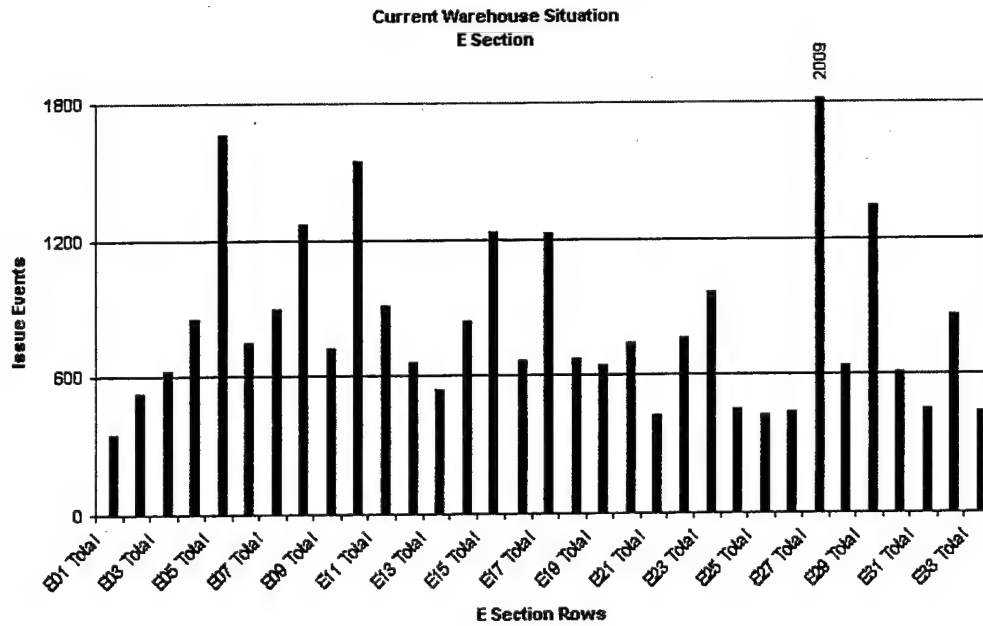


Figure 8. Section E issues by row before stock relocation with the Item Location Assignment Algorithm

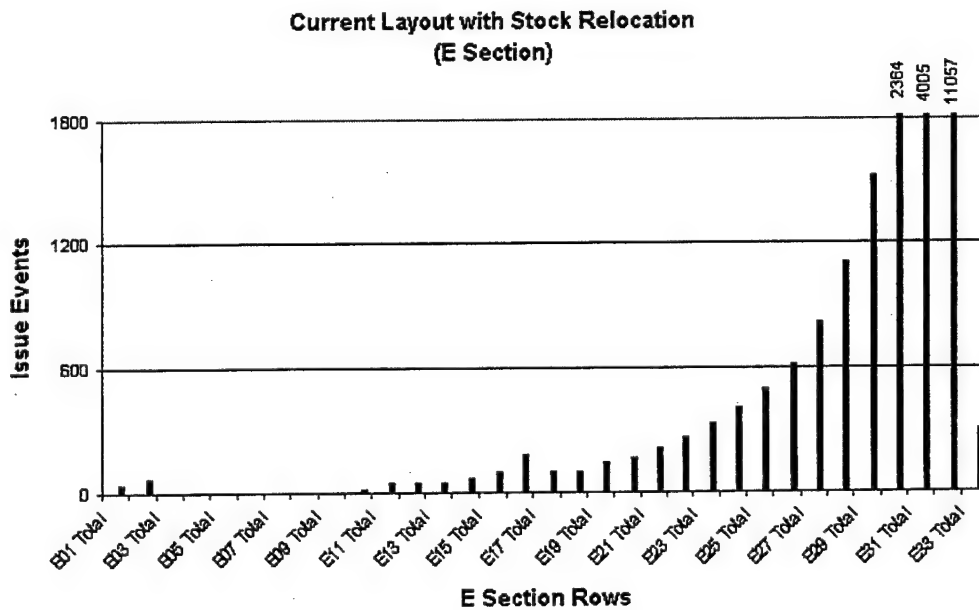


Figure 9. Section E issues by row after relocating stock with the Item Location Assignment Algorithm

in Section H, an overall trend of items being located closer to the I/O point is still clearly visible.

Total estimated travel distance over five quarters for the current layout is 1,283,734 feet for forklift items and 2,480,779 feet for the batch picked items. These distances reduce to 590,455 feet for forklift items and 1,661,480 feet for batch picked items when we employ the Item Location Assignment Algorithm. The relocation of items results in a reduction of 40.2% in expected travel distance.

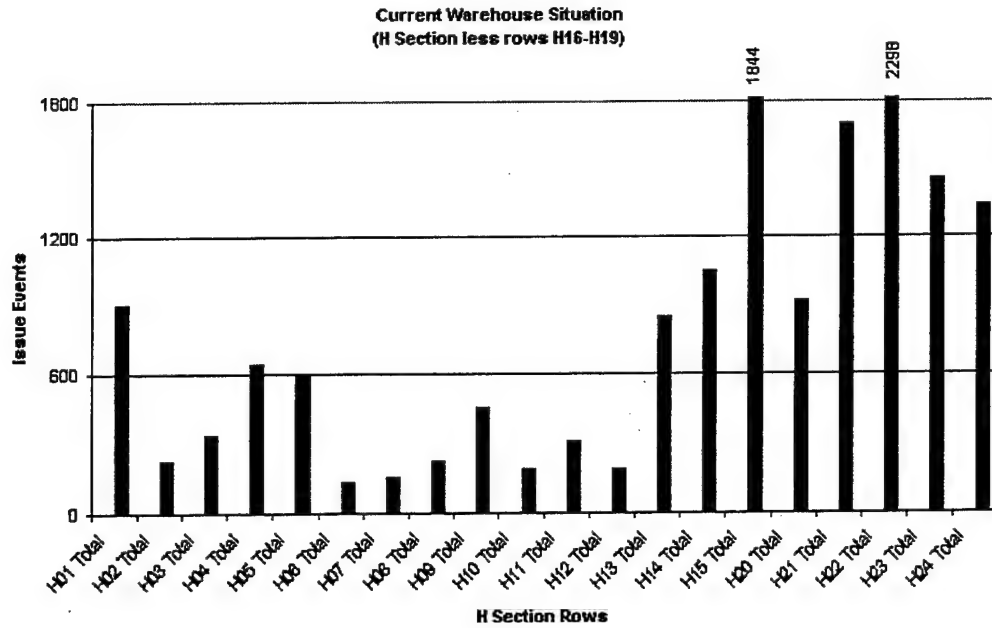


Figure 10. Section H issues by row before stock relocation with the Item Location Assignment Algorithm

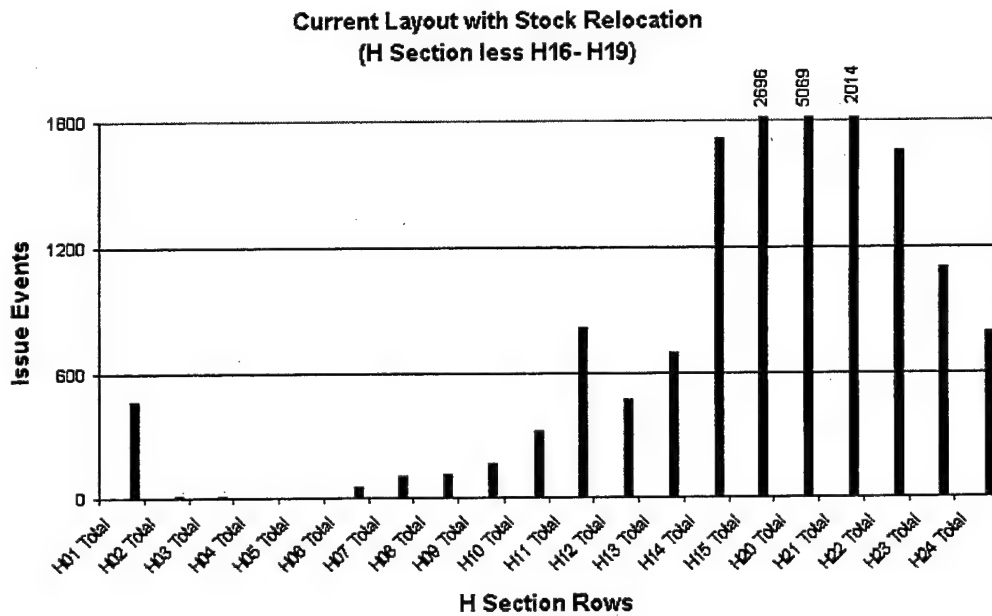


Figure 11. Section H issues by row after relocating stock with the Item Location Assignment Algorithm

IV. RECONFIGURING STORAGE DEVICES

A. A PAIRWISE-INTERCHANGE ALGORITHM

We determine an improved configuration of the storage devices by using a Pairwise-interchange Algorithm. Since ARSC has a budget limit on what they can invest to improve their layout, we determine the greatest reduction in expected travel distance within their specified 7 rack relocation limit. ARSC selected 7 as their limit based on fiscal constraints. They estimated that it would cost approximately \$136K to reconfigure 7 devices. This is what they are willing to invest to reconfigure the warehouse contingent upon the results of our study. The algorithm makes limited adjustments to the warehouse's current configuration by swapping the positions of storage devices. In order to maintain aisle widths for adequate vehicle passage, the devices that can be swapped are severely limited. Because the devices in which forklift items are stored require such a wide aisle, we are not able to swap *blue*, *pink* or *brown* devices with any other device but they may be swapped among themselves. We may swap *orange* devices with *yellow* devices and vice-versa. Six rows of *green* devices may be swapped with four rows of *orange* or *yellow* devices or we may swap rows B21 & B22 or B23 & B24 with E33 & H01 and still maintain adequate aisle widths.

1. Algorithm

Our solution method involves a two-part algorithm. In the first part we fix storage devices and assign items to optimal locations with the Item Location Assignment Algorithm. We assign items to locations before storage devices because we believe the devices are closer to being in good locations than the items. This part of the algorithm optimally matches the highest usage items to the closest feasible locations (with respect to device types).

In the second part we make a greedy selection of racks for interchange, check for physical feasibility of the interchange, make the interchange, reassign item locations within the new configuration and check for improvement in reduced travel distances.

Since fiscal constraints limit the number of swaps that we can make, we swap only a limited number (7) of racks. In order to compare racks to determine interchanges, we normalize their issue events by dividing the number of issue events for each rack by the number of items picked in a batch for the rack's device type. We choose not to use a meta-heuristic such as simulated annealing because ARSC is willing to entertain so few relocations. The algorithm is:

Algorithm: Greedy Pairwise-Interchange

Input: item matrix (NIIN, location, issue events)

device matrix (row position, device type)

stopLimit variable (limits interchanges)

Output: new item matrix, new device matrix

Begin:

scalar forkDistance \leftarrow 0;

scalar aisleDistance \leftarrow 0;

scalar weightedDistance \leftarrow 0;

scalar newDistance \leftarrow 0;

scalar improvedDistance \leftarrow 0;

scalar stopLimit \leftarrow n;

scalar counter \leftarrow 0;

// Determine Interchanges

run the Item Location Assignment Algorithm;

boolean done \leftarrow false;

while (not.done);

calculate normalized usage for each row;

A \leftarrow rack with greatest normalized usage;

B \leftarrow interchangeable rack with least normalized
usage closer to I/O than A;

if there is no B for selected A then:

while there is no B for selected A do:

A \leftarrow rack with next greatest normalized usage;

B \leftarrow interchangeable rack with least
normalized usage closer to I/O than A;

end while;

end if;

interchange A and B;

run the Item Location Assignment Algorithm;

improvedDistance \leftarrow newDistance

counter = counter + 1;

if counter = stopLimit then;

```

done ← true;
end if;
end while;
end algorithm

```

We implement the algorithm in Microsoft Excel 97 with Visual Basic Applications (VBA).

2. Results

Figure 12 shows the results of the interchanges. Total expected travel distance for forklift items after the 7 interchanges remains 590,455 feet (no forklift item devices were interchanged) and distance for batch picked items is 1,451,351 feet. This is an additional reduction of 5.6% over simply relocating items within ARSC's current layout, resulting in a total 45.8% reduction in travel distance over the current layout.

B. AN ITERATIVE LINEAR PROGRAMMING ALGORITHM

Here we develop a general method for locating different types of storage devices in a warehouse and allocating items for them. Although the solution cannot be implemented by ARSC because of budget constraints, our results provide a lower bound on possible improvements due to warehouse redesign. Furthermore, the algorithm could be used by other DOD or commercial activities.

1. Algorithm

We begin by calculating the cross aisle distance from the I/O point to each rack. We then use the Item Location Assignment Algorithm to optimally assign items to locations for the given layout. With each item in its optimal location, we calculate the normalized usage for each rack. The LP assigns racks to the locations. Once a new rack assignment is determined we check for a decrease in objective function value. If the objective function value decreases, we calculate new normalized usages based on the solution to the LP and repeat the process. If no improvement is made

**Pairwise-
Interchange
Solution**

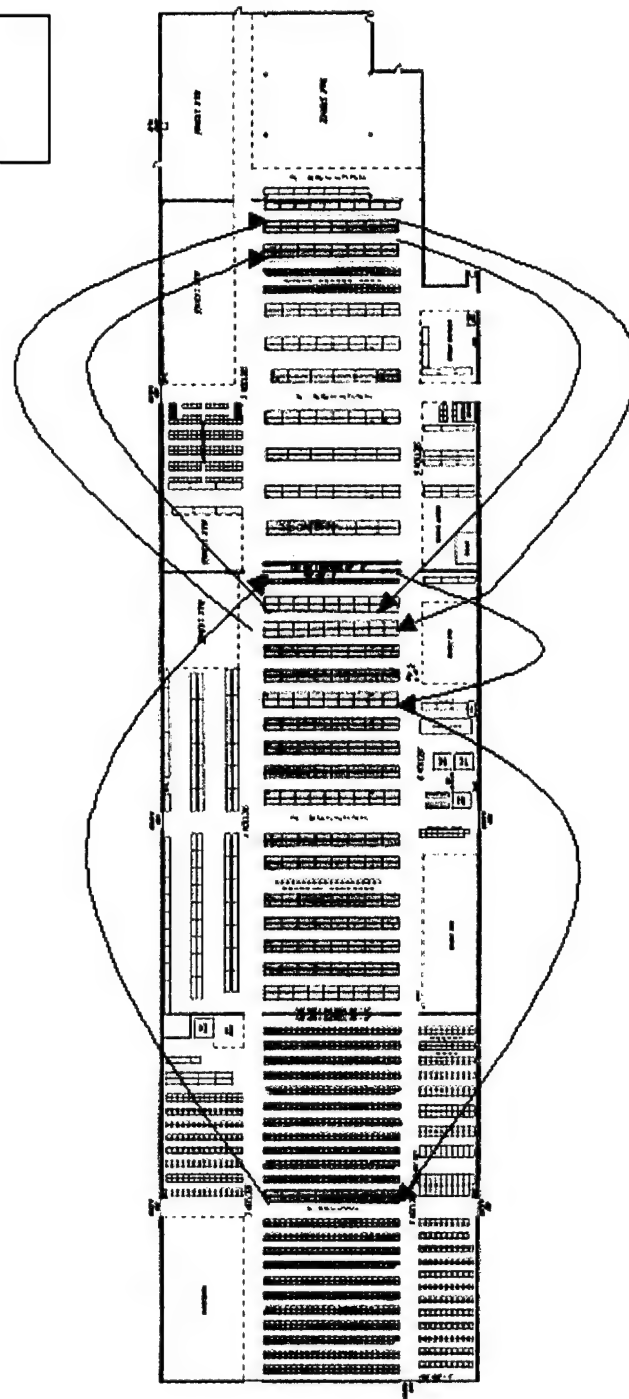


Figure 12. The improved layout of the ARSC warehouse with storage devices color coded. Arrows indicate interchanged racks. Arrows on the left show racks that move forward. Arrows on the right show racks that move toward the rear of the warehouse.

we adjust the aisle widths to ensure physical feasibility for the material handling equipment and the warehouse walls. We use the following iterative algorithm:

Algorithm: Iterative Linear Program

Begin:

1. Calculate cross aisle distance d from the I/O point to each rack.
 2. Run the Item Location Assignment Algorithm.
 3. Compute normalized usage for each rack.
 4. Solve a Linear Programming optimization model.
 5. If cost decreases go to Step 2; else go to Step 6.
 6. Adjust aisle widths to ensure physical feasibility.
- end algorithm**

Notation for the model is:

Indices:

- i rack number,
 j location of rack.

Data:

- c_i estimated number of trips to rack i ,
 d_j cross aisle distance of location j from I/O point.

Variables:

$$X_{ij} \begin{cases} 1 & \text{if rack } i \text{ is assigned to location } j, \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

The model is

$$\text{Minimize } \sum_i \sum_j c_i d_j X_{ij}$$

$$\text{subject to } \sum_j X_{ij} = 1 \quad \forall i \quad (\text{IV.1})$$

$$\sum_i X_{ij} = 1 \quad \forall j \quad (\text{IV.2})$$

$$X_{ij} \in \{0,1\} \quad \forall i,j. \quad (\text{IV.3})$$

The objective function minimizes the product of the cross aisle distance to the rack's location and the cost of locating the rack from the I/O point. Constraint set IV.1 ensures that each rack has only one location. Constraint set IV.2 ensures that each location is assigned only one rack. Since the model is an assignment problem, the LP relaxation gives an integer solution.

We implement our algorithm in Microsoft Excel 97 with Visual Basic Applications (VBA). We use the Premium Solver Plus Version 3.5 for Excel add-in from Frontline Systems to solve the linear programming assignment problem. Premium Solver Plus is an upwardly compatible extension of the standard Microsoft Excel Solver (SOLVER.XLA) and has the capacity to solve linear problems with up to 2000 variables (Terry, 1999). The ARSC warehouse has 53 aisles. After running the Item Location Assignment Algorithm, we determine that only 41 aisles have active inventory. We are therefore able to reduce the number of variables for our linear programming problem from $53^2 = 2809$ variables to only $41^2 = 1681$ variables and remain within the capacity of the Premium Solver Plus. The solution for the Item Location Assignment portion takes approximately 6 minutes per iteration on a PC with a Cyrix 6x86 P200+ microprocessor (Intel 200 Mhz Pentium Pro processor equivalent). The LP assignment problem portion requires less than six seconds to solve per iteration. The Iterative LP Algorithm finds the best solution in seven iterations taking approximately 43 minutes.

2. Results

Figure 13 shows the results of the Iterative Linear Programming Algorithm. Total expected travel distance is 495,662 feet for forklift items and 1,246,296 feet for batch picked items. This is an additional 7.9% reduction over the Pairwise-interchange Algorithm and a total reduction in travel distance of 53.7% over ARSC's current layout. The solution provides a location for at least one rack of each device type near the I/O point, but requires the relocation of every rack in the warehouse. Racks of the same device type with the same normalized usage are grouped together. For example,

the *green* and *orange* racks depicted at the rear of the warehouse in Figure 13 all have zero issue events. Unfortunately, space for 2.5 *green* racks is lost due to increased space allocated to aisle widths. More space must be allocated to aisle widths because of the loss of uniformity of device types within the Sections. Additionally, overhead lighting location requirements also change as the aisles are no longer located beneath lights.

LP
Best
Layout

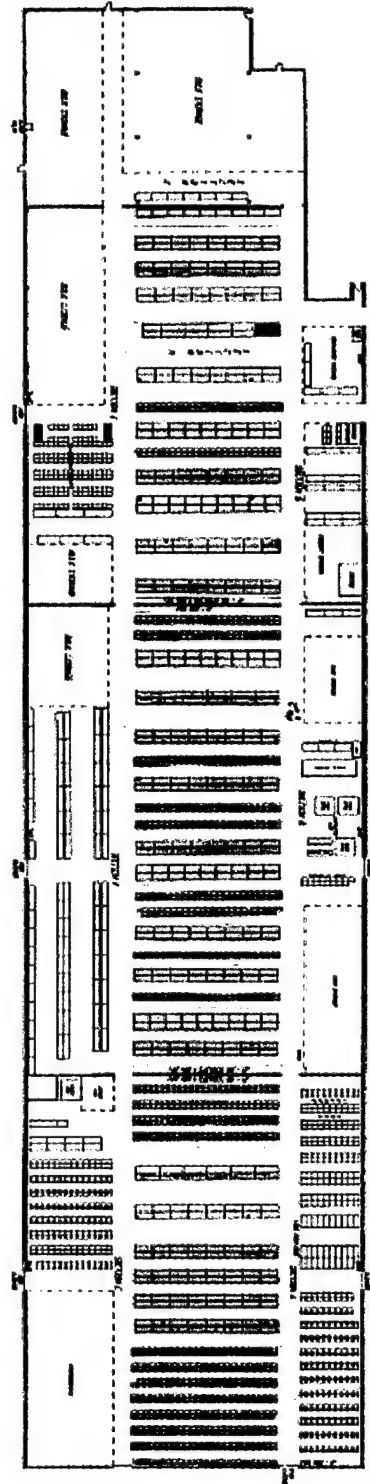


Figure 13. The best layout of the ARSC warehouse with storage devices color coded.

V. SUMMARY AND CONCLUSIONS

Warehouse design and subsequent rewarehousing are strategic decisions since they have a long-term impact on labor costs and profitability (Cormier and Gunn, 1992). Once we implement these decisions, it is expensive and difficult to change them. This thesis addresses the problem of finding an improved warehouse layout to reduce the distances workers travel to pick items on an order. We produce three models that decrease expected travel distances over the current layout. The first model relocates items within the current layout, the second finds an improved layout by interchanging a limited number of storage devices, and the third determines the best layout regardless of device relocation limitations.

To interpret our results, we convert total expected travel distance into dollar savings for ARSC's current operating procedure and for each of our three solutions. We convert expected travel distances to dollars saved by scaling our results to one year, then converting to seconds per year spent travelling, and finally to dollars saved per year by multiplying by the man-hour item picking costs. We summarize our calculations in Table 2,

	Current Operating Procedures		Item Location Assign Algorithm		Pairwise-interchange Algorithm		Iterative LP Algorithm	
	fork lifts	batch pick	fork lifts	batch pick	fork lifts	batch pick	fork lifts	batch pick
Travel Distances	1,283,734	2,480,779	590,455	1,661,480	590,455	1,451,351	495,662	1,246,296
Hours/Year	47.54	64.33	21.86	43.08	21.86	37.63	18.35	32.31
Dollars/Year	\$5,431	\$7,349	\$2,498	\$4,922	\$2,498	\$4,299	\$2,097	\$3,692
Total Pick Cost	\$12,780		\$7,420		\$6,797		\$5,789	
Dollars Saved	————		\$5,360		\$5,983		\$6,991	

Table 2. Results for ARSC's current operating procedure as well as for each of our three solution methods.

When we compare the dollars saved with the costs to make a device layout configuration change to the warehouse, it is clear that changing the layout of the storage devices is not cost effective.

The ARSC warehouse is operating in a steady state condition, by which we mean the aircraft being supported are all mature, and the distribution of parts being ordered is unlikely to change. The ARSC warehouse will remain in this state until there is a new aircraft acquisition or retirement of an aircraft type. We anticipate that the current aircraft types will remain unchanged for at least eight to ten more years. By relocating current stock with the results from the Item Location Assignment Algorithm, ARSC can achieve a 40.2% reduction in annual labor costs to pick items, saving approximately \$53,600 in labor costs in 1999 dollars over the next ten years. There would be additional savings if activity at the warehouse increases in the coming years, as managers currently expect.

There will also be additional savings in labor cost because distances travelled to stock items will also be reduced. We do not have order quantity information from ARSC and so are not able to estimate this additional savings. (Since items are often stocked by a lot size order, but issued in a smaller quantity, the savings associated with reduced travel distances to stock items would be less than those to pick items.) The labor hours saved may be used to absorb the requirements of an increased work load or they may be reprogrammed into other warehouse activities such as housekeeping, inventory accuracy, as well as achieving shorter cycle times for delivery of items to the Air Stations making orders.

We recommend that ARSC implement the results from the Item Location Assignment Algorithm within their current storage device layout. ARSC may efficiently implement the solution by relocating items during an inventory when the warehouse is otherwise closed to issue and receipt activity. ARSC may accomplish relocations for items stored in *green* devices by using the vacant A & B bins as temporary holding areas. Since there are no vacant bins for the other device types, a temporary holding area must be established while relocating items. The I/O point is free of activity during an inventory and would serve as an ideal temporary holding area for the items in non-*green* device types. We recommend ARSC personnel relocate the forklift items

first. These items provide the greatest reduction in travel distances. We recommend the items in *orange* devices assigned to rows E32, E31, E30, and E29 and the items in *yellow* devices assigned to rows H20, H22, and H23 be relocated next since they contain the fastest moving large batch picked items. Finally the items in *green* devices assigned to rows H16–H19 should be relocated since they contain the fastest moving small batch picked items. This method of allocating effort provides the greatest margin of return for reducing travel distances since these rows contain the bulk of the issue activity after reassignment of items. After completing the reassignment of items in these rows, ARSC may relocate the remaining items as time becomes available.

APPENDIX A.

The Item Location Assignment Algorithm is:

Algorithm: Item Location Assignment

Input: item matrix (NIIN, location, issue events, b_i , q_i , l , m)
device matrix (row position, device type)

Output: new item matrix, new device matrix

Begin:

```
    scalar forkDistance  $\leftarrow$  0;
    scalar crossAisleDist  $\leftarrow$  0;
    scalar inAisleDist  $\leftarrow$  0;
    scalar aisleDistance  $\leftarrow$  0;
    scalar totalDistance  $\leftarrow$  0;
    scalar newDistance  $\leftarrow$  0;
// Compute total distances
    calculate current location and aisle distances;
    vector location  $\leftarrow$  location distances;
    vector aisle  $\leftarrow$  aisle distances;
    for each device type do;
        for each forklift item do;
            forkDistance  $\leftarrow$  forkDistance +
                (location * issue events);
        end for;
        for each batch pick aisle do;
            compute E[crossAisleDist];
            crossAisleDist  $\leftarrow$  crossAisleDist + E[crossAisleDist];
            compute  $Q_j$ ;
            compute  $P_j$ ;
        end for;
        compute expected in-aisle distance (inAisleDist);
        aisleDistance  $\leftarrow$  2(crossAisleDist) + inAisleDist;
        totalDistance  $\leftarrow$  forkDistance + aisleDistance;

// Solve the item location problem
    for each device type do;
        order items by issue events in decreasing issue events;
        if device is forklift picked then;
            order location distances in increasing distance;
            for each forklift item do;
                do not assign bins "A" or "B" until all other
                    bins are assigned;
```



```

        if item requires more then one location then;
            assign locations for item in the same
                row and in adjacent bins;
        end if;
        assign item with highest usage to location
            with shortest distance;
    end for;
end if;
if device is batch picked then;
    order aisle distances in increasing distance;
    for each batch pick item do;
        if device type is green then;
            do not assign bins "A" or "B";
            if item requires > one location then;
                assign locations for item in the
                    same row & in adjacent bins;
            end if;
            assign item with highest usage to aisle
                with shortest distance;
        else (device type is not green) then;
            do not assign bins "A" or "B" until
                all other bins are assigned;
            if item requires > one location then;
                assign locations for item in the
                    same row & in adjacent bins;
            end if;
            assign item with highest usage to aisle
                with shortest distance;
        end if;
    end for;
end if;
end for;
forkDistance  $\leftarrow$  0;
for each forklift item do;
    forkDistance  $\leftarrow$  forkDistance + (location * issue events);
end for;
aisleDistance  $\leftarrow$  0;
for each batch pick aisle do;
    compute E[crossAisleDist];
    crossAisleDist  $\leftarrow$  crossAisleDist + E[crossAisleDist];
    compute  $Q_j$ ;
    compute  $P_j$ ;

```

```
    end for;  
    compute expected in-aisle distance (inAisleDist);  
    aisleDistance  $\leftarrow$  2(crossAisleDist) + inAisleDist;  
    newDistance  $\leftarrow$  forkDistance + aisleDistance;  
end algorithm
```


LIST OF REFERENCES

- Ackerman, Kenneth B., *The Practical Handbook of Warehousing*, 3rd ed., Van Nostrand Reinhold, New York, 1990.
- Ashayeri, J. and Gelders, L.F., "Warehouse Design Optimization," *European Journal of Operations Research*, v. 21, pp. 285-294, 1985.
- Burgess, Ken. CDR, USCG, Personal Interview, November 1998.
- Cormier, Gilles and Gunn, Eldon A., "A Review of Warehouse Models," *European Journal of Operations Research*, v. 58, pp. 3-13, 1992.
- Francis, Richard L., McGinnis Jr., Leon F., and White, John A., *Facility Layout and Location: An Analytical Approach*, 2nd ed., Prentice Hall, Englewood Cliffs, New Jersey, 1992.
- Francis, Richard L., McGinnis, Leon F., and White, John A., "Locational Analysis," *European Journal of Operations Research*, v. 12, pp. 220-252, 1983.
- Gibson, David R. and Sharp, Gunter P., "Order Batching Procedures," *European Journal of Operations Research*, v. 58, pp. 61-62, 1992.
- Gray, Ann E., Karmarkar, Uday S., and Seidmann, Abraham, "Design and Operation of an Order Consolidation Warehouse: Models and Application," *European Journal of Operations Research*, v. 58, pp. 14-36, 1992.
- Guenov, Marin and Raeside, Robert, "Zone Shapes in Class Based Storage and Multicommand Order Picking when Storage/Retrieval Machines Are Used," *European Journal of Operations Research*, v. 58, pp. 37-47, 1992.
- Harmatuck, Donald J., "A Comparison of Two Approaches to the Stock Location Problem," *The Logistics Transportation Review*, v. 12, pp. 282-284, 1976.
- Hausman, Warren H., Schwarz, Leroy B., and Graves, Stephen C., "Optimal Storage Assignment in Automatic Warehousing Systems," *Management Science*, v. 22(6), pp. 629-638, 1976.
- Heragu, Sunderesh, *Facilities Design*, PWS Publishing Company, Boston 1997.
- Heskett, J.L., "Cube-Per-Order Index - A Key to Warehouse Stock Location," *Transportation and Distribution Management*, v. 3, pp. 27-31, 1963.
- Hudock, Brian J., "Design by Philosophy," *Transportation and Distribution*, pp.127-128, May 1996.

Jarvis, Jay M. and McDowell, Edward D., "Optimal Product Layout in an Order Picking Warehouse," *IIE Transactions*, v. 23(1), pp. 93-97, 1991.

Kallina, Carl and Lynn, Jeffrey, "Application of the Cube-Per-Order Index Rule for Stock Location in a Distribution Warehouse," *Interfaces*, v. 7(1), pp. 37-46, 1976.

McCarty, Robert, LCDR, USCG and Limbacher, Paul, LT, USCG Various E-mails between McCarty and Limbacher, ARSC and LCDR J. F. Martin, through April 1999.

Nelson, Raymond A., *Computerizing Warehouse Operations*, Prentice Hall, Englewood Cliffs, New Jersey, 1985.

Terry, P., E-mail between P. Terry, Sales Manager, Frontline Systems Inc., and LCDR J. F. Martin, May 1999.

Tompkins, James A. and Smith, Jerry D., *The Warehouse Management Handbook*, McGraw-Hill, Inc., New York, 1988.

Tompkins, James A. and White, John A., *Facilities Planning*, John Wiley & Sons, Inc., New York, 1984.

Wilson, Hoyt G., "Order Quantity, Product Popularity, and the Location of Stock in a Warehouse," *AIIE Transactions*, v. 9(3), pp. 230-237, 1977.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center2
 8725 John J. Kingman Rd., STE 0944
 Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library2
 Naval Postgraduate School
 411 Dyer Rd.
 Monterey, CA 93943-5101

3. Commandant1
 U.S. Coast Guard (G-SEA)
 2100 2nd Street SW
 Washington, DC 20593-0001

4. Commandant1
 U.S. Coast Guard (USCG Law Library)
 2100 2nd Street SW
 Washington, DC 20593-0001

5. Commanding Officer1
 U. S. Coast Guard
 Aircraft Repair and Supply Center
 Building 63
 Elizabeth City, NC 27909

6. James F. Martin1
 8324 Briar Creek Dr
 Annandale, VA 22003

7. Dr. Kevin R. Gue2
 Systems Management Department (Code SM/Gk)
 Naval Postgraduate School
 555 Dyer Rd.
 Monterey, CA 93943-5103

8. Dr. Gordon H. Bradley1
 Operations Research Department (Code OR/Bz)
 Naval Postgraduate School
 1411 Cunningham Rd.
 Monterey, CA 93943-5219